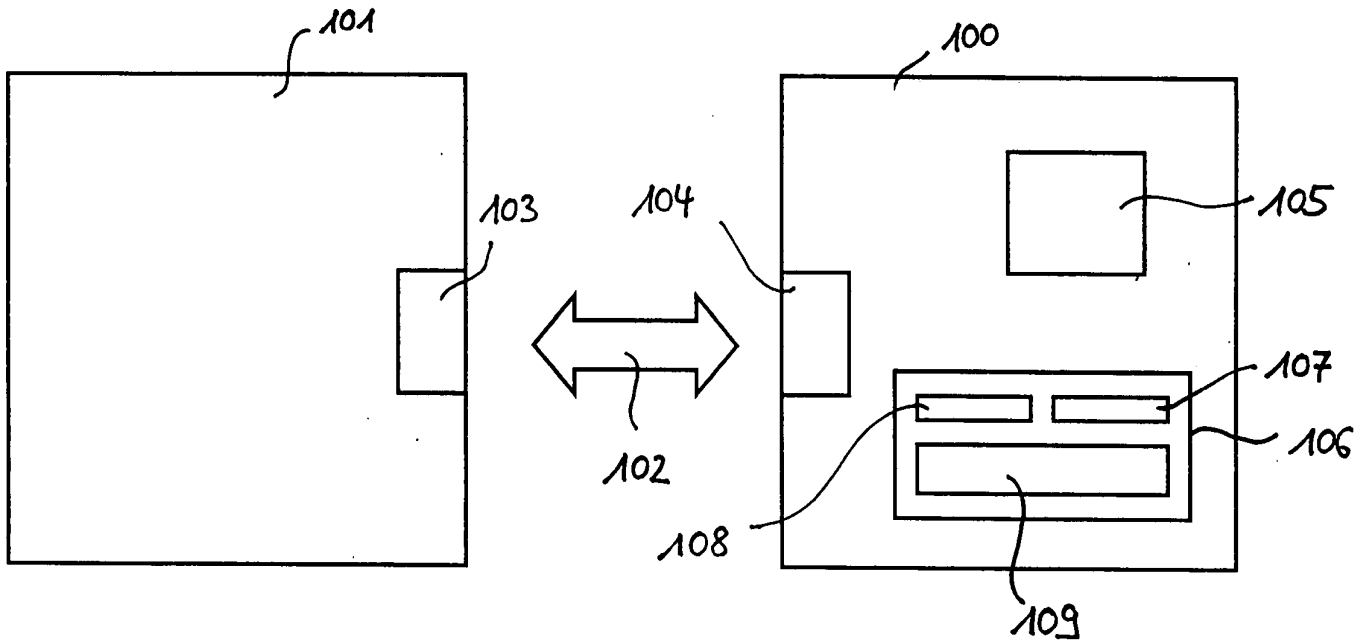


Fig. 1

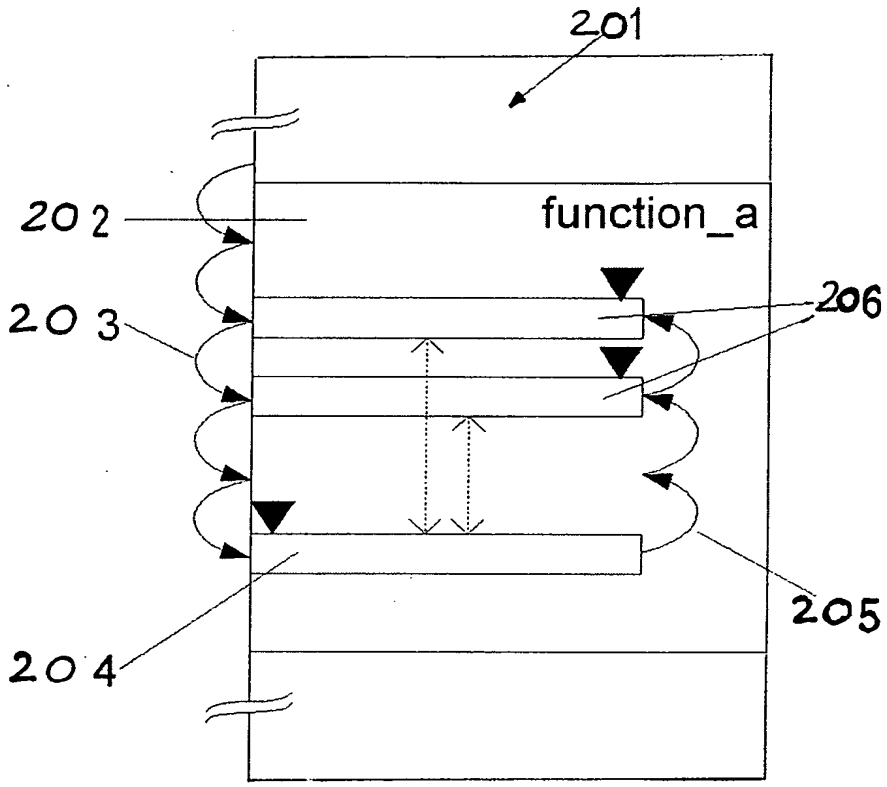


Fig. 2

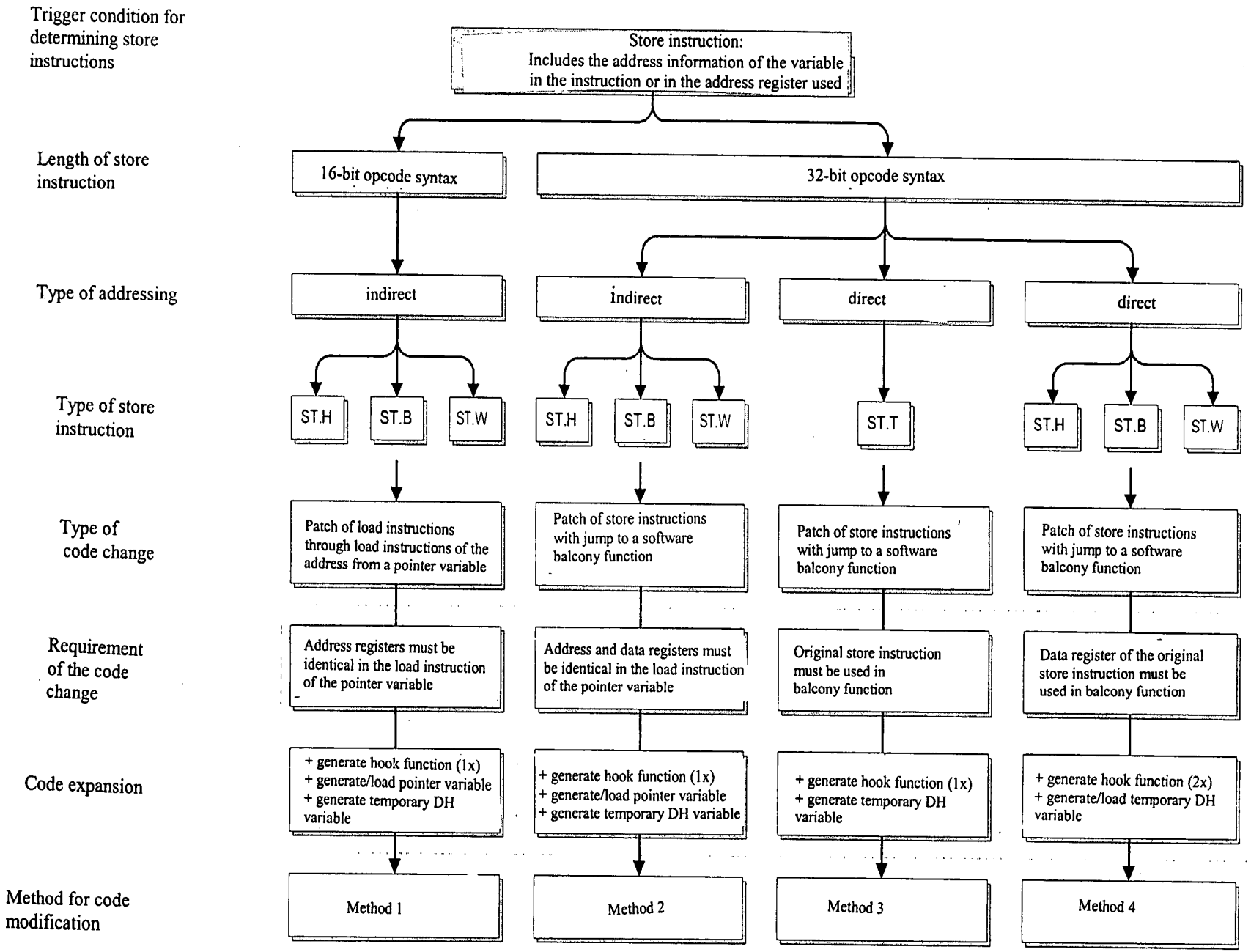

Trigger condition for determining store instructions
Store instruction:
Includes the address information of the variable in the instruction or in the address register used
Length of store instruction
16-bit opcode syntax
32-bit opcode syntax
Type of addressing
indirect
indirect
direct
direct
Type of store instruction
ST.H
ST.B
ST.W
ST.H
ST.B
ST.W
ST.T
ST.H
ST.B
ST.W
Type of code change
Patch of load instructions through load instructions of the address from a pointer variable
Patch of store instructions with jump to a software balcony function
Patch of store instructions with jump to a software balcony function
Patch of store instructions with jump to a software balcony function
Requirement of the code change
Address registers must be identical in the load instruction of the pointer variable
Address and data registers must be identical in the load instruction of the pointer variable
Original store instruction must be used in balcony function
Data register of the original store instruction must be used in balcony function
Code expansion
+ generate hook function (1x)
+ generate/load pointer variable
+ generate temporary DH variable
+ generate hook function (1x)
+ generate/load pointer variable
+ generate temporary DH variable
+ generate hook function (1x)
+ generate temporary DH variable
+ generate hook function (2x)
+ generate/load temporary DH variable
Method for code modification
Method 1
Method 2
Method 3
Method 4

FIG. 3

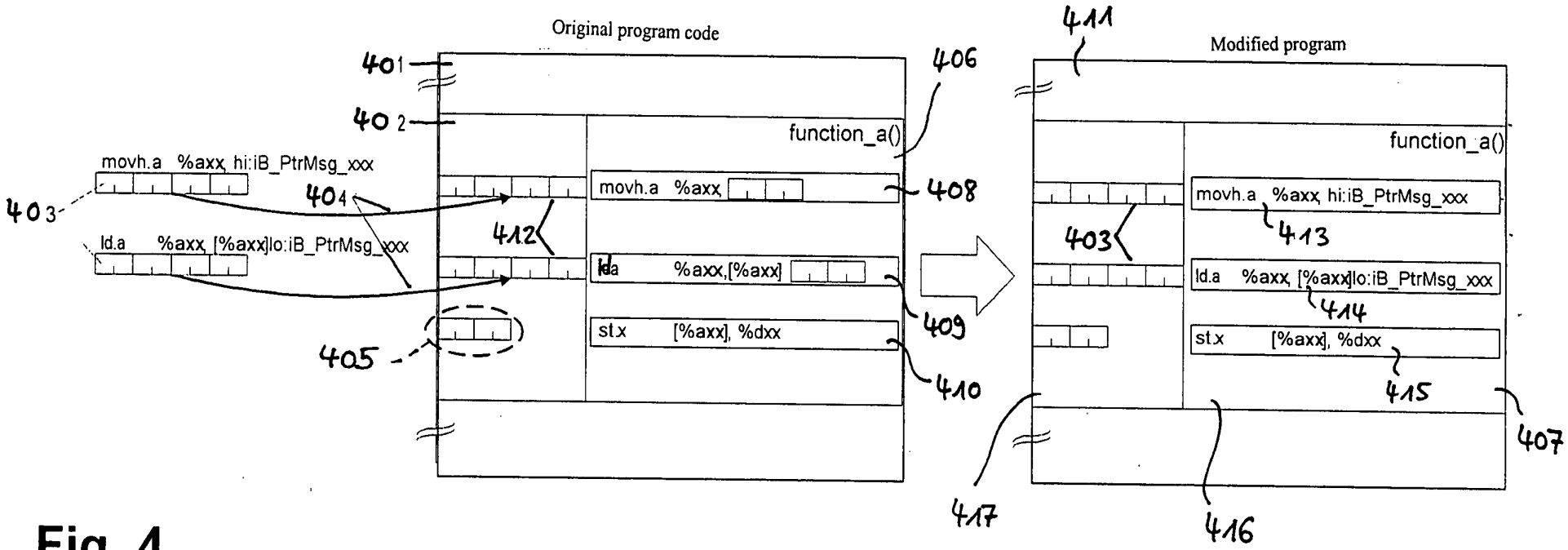
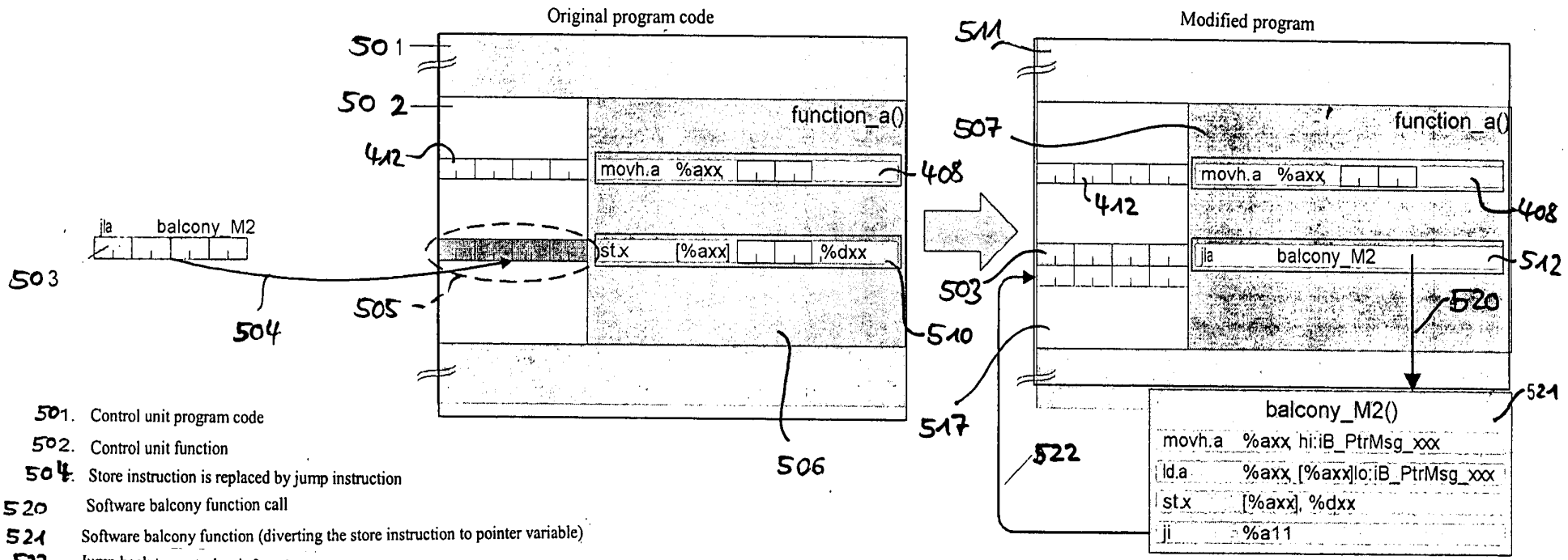

Original program code
401
402
406
function_a()
movh.a %axx, hi:iB_PtrMsg_xxx
403
404
movh.a %axx,
408
412
ld.a %axx, [%axx]lo:iB_PtrMsg_xxx
ld.a %axx,[%axx]
409
405
st.x [%axx], %dxx
410
411
Modified program
function_a()
movh.a %axx, hi:iB_PtrMsg_xxx
413
403
ld.a %axx, [%axx]lo:iB_PtrMsg_xxx
414
st.x [%axx], %dxx
415
407
417
416

Fig. 4



501. Control unit program code

502. Control unit function

504. Store instruction is replaced by jump instruction

520 Software balcony function call

521 Software balcony function (diverting the store instruction to pointer variable)

522 Jump back to control unit function

st.x = st.b (store byte), st.h (store half word), st.w (store word), ...

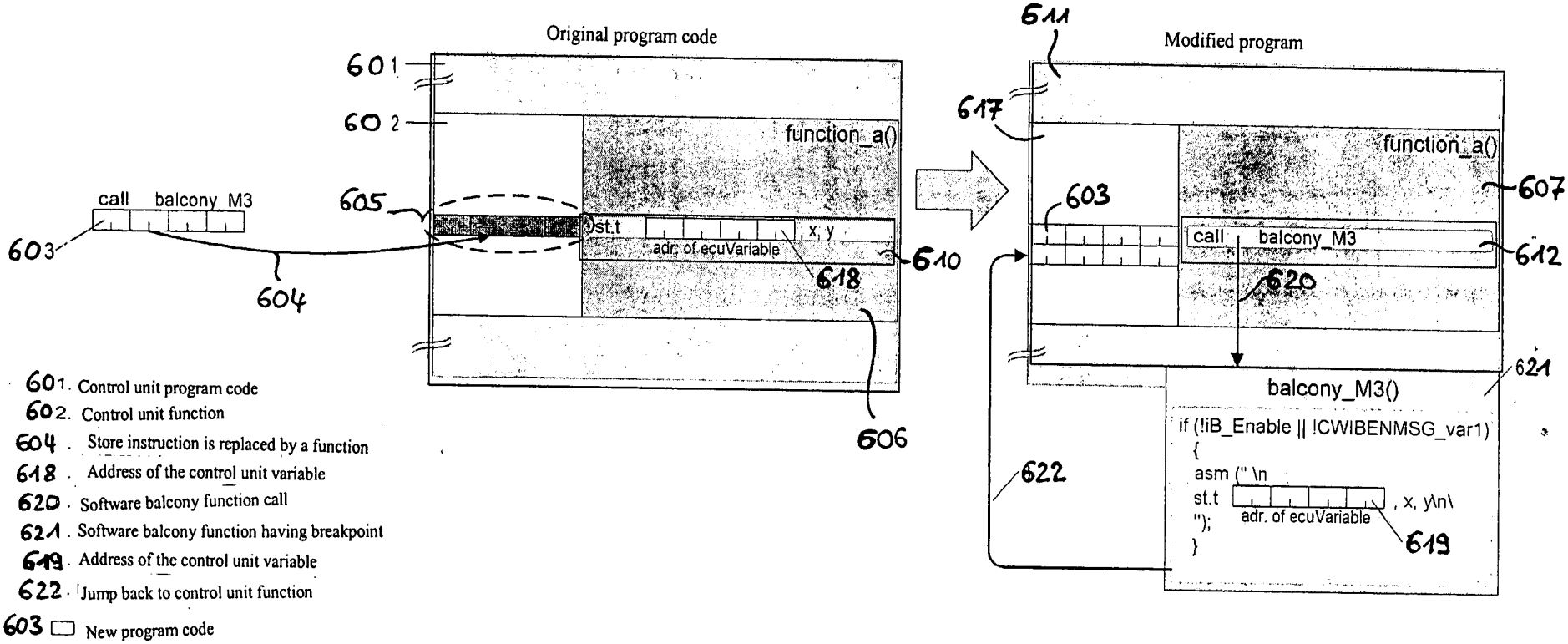
axx = Address register a0 ... a15

dxx = Data registers d0 ... d15

503 New program code

Fig. 5

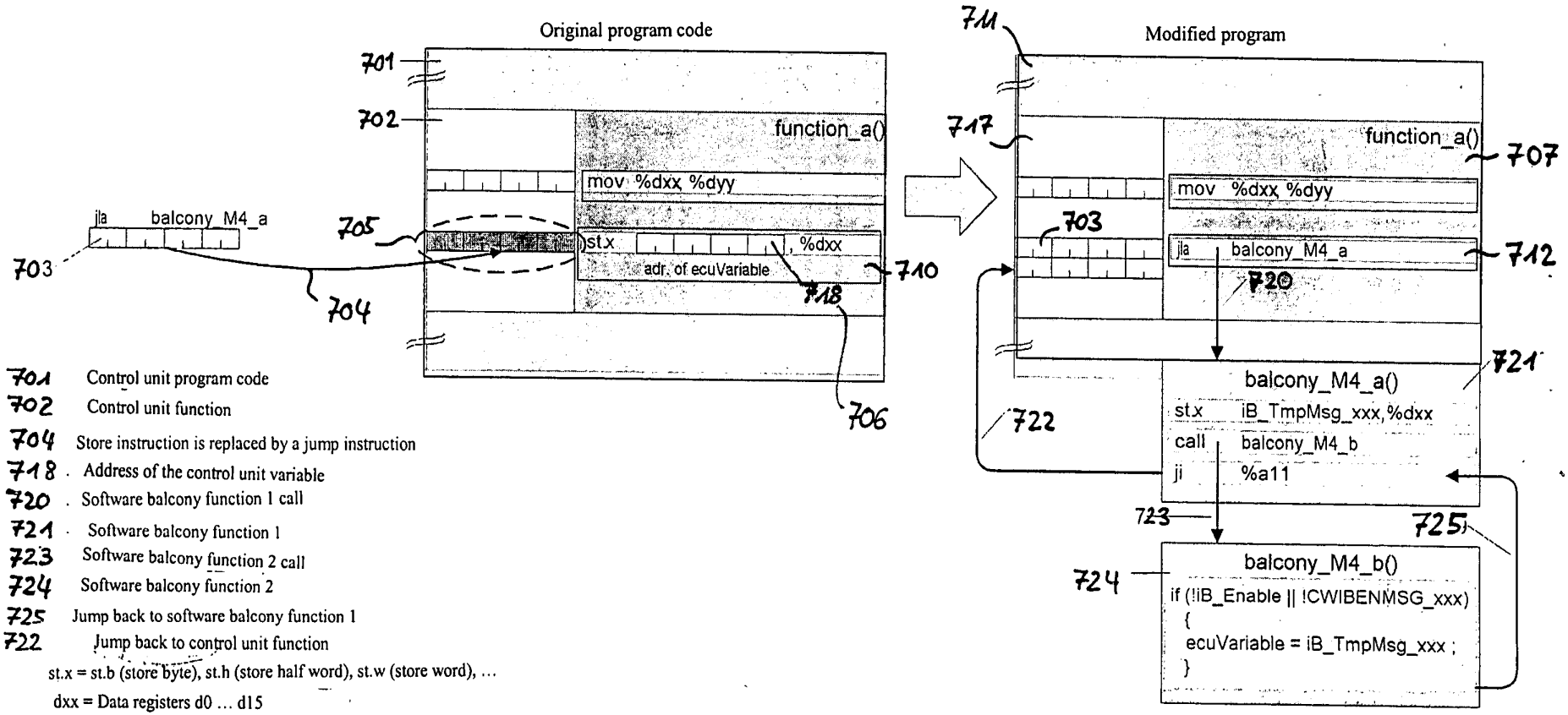
BEST AVAILABLE COPY



601. Control unit program code
602. Control unit function
604. Store instruction is replaced by a function
618. Address of the control unit variable
620. Software balcony function call
621. Software balcony function having breakpoint
619. Address of the control unit variable
622. Jump back to control unit function
603 New program code

Fig. 6

BEST AVAILABLE COPY



701 Control unit program code
702 Control unit function
704 Store instruction is replaced by a jump instruction
718 Address of the control unit variable
720 Software balcony function 1 call
721 Software balcony function 1
723 Software balcony function 2 call
724 Software balcony function 2
725 Jump back to software balcony function 1
722 Jump back to control unit function

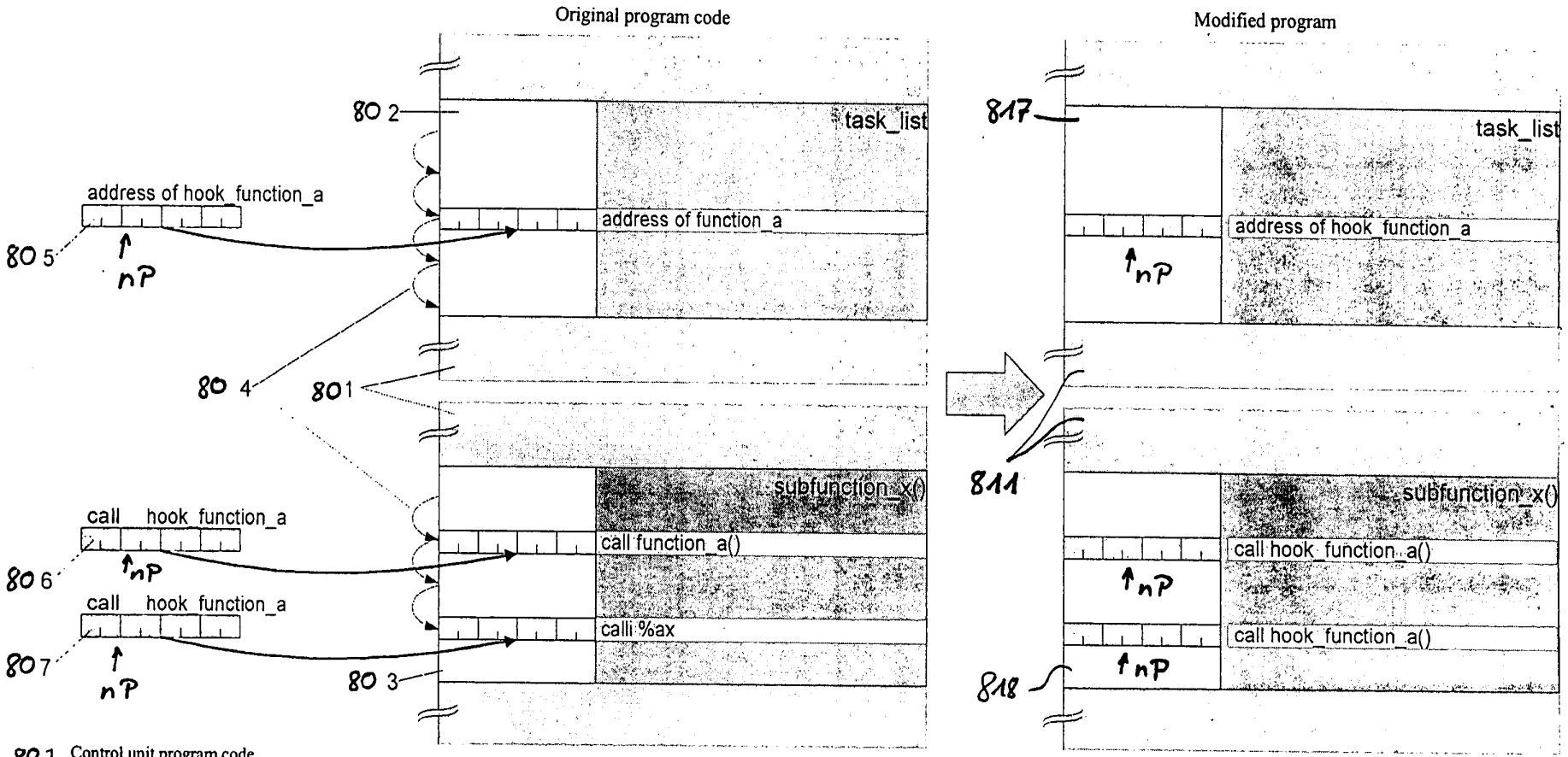
st.x = st.b (store byte), st.h (store half word), st.w (store word), ...

dxx = Data registers d0 ... d15

703 New program code

Fig. 7

BEST AVAILABLE COPY



801. Control unit program code
802. Task list
803. Subfunction
804. Procedure for determination of function addresses and function calls
805. Address of function_a is replaced by address of hook_function_a
806. Function call of function_a is replaced by call of hook_function_a
807. Indirect function call of function_a is replaced by call of hook_function_a (32-bit instruction

nP New program code

Fig. 8

BEST AVAILABLE COPY

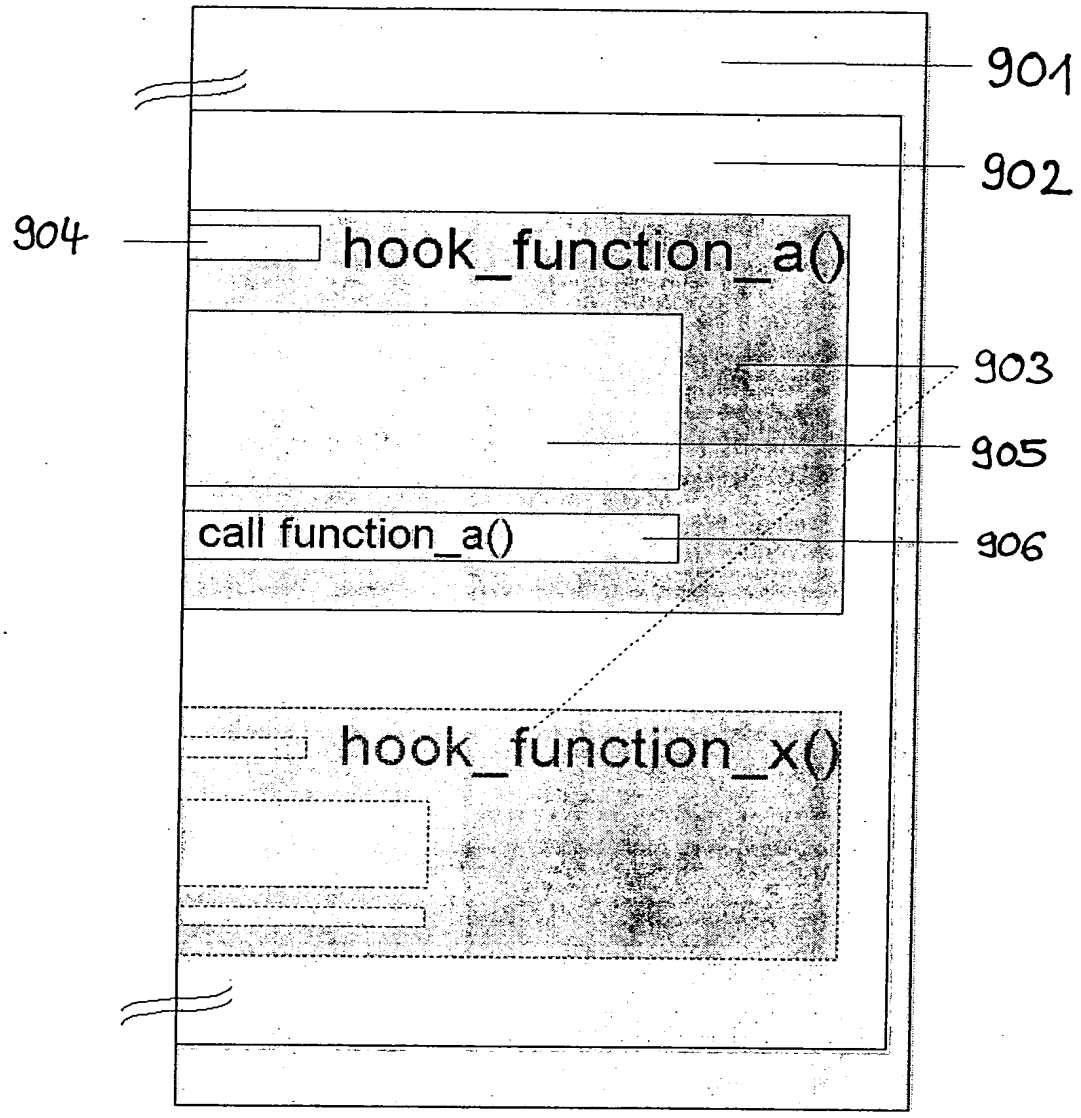


Fig. 9

BEST AVAILABLE COPY

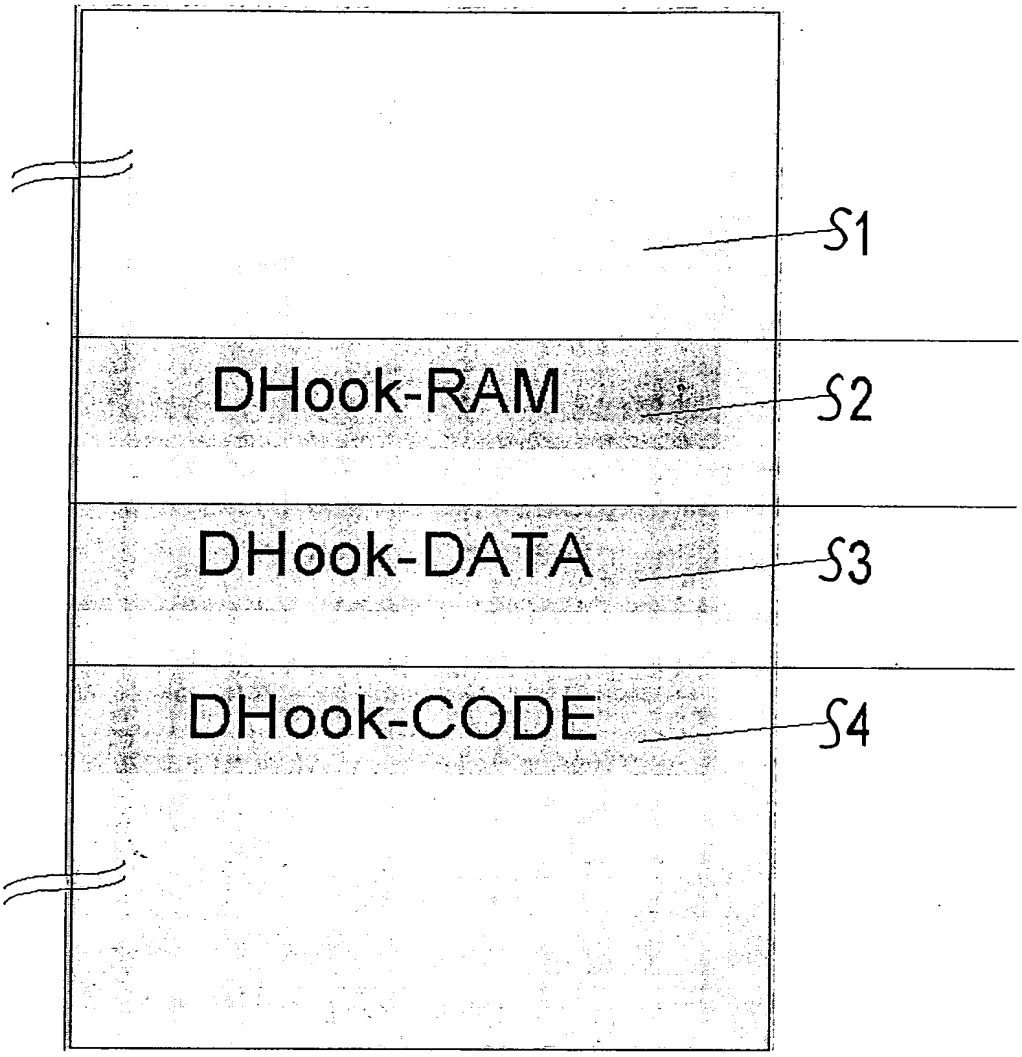


Fig. 10

BEST AVAILABLE COPY

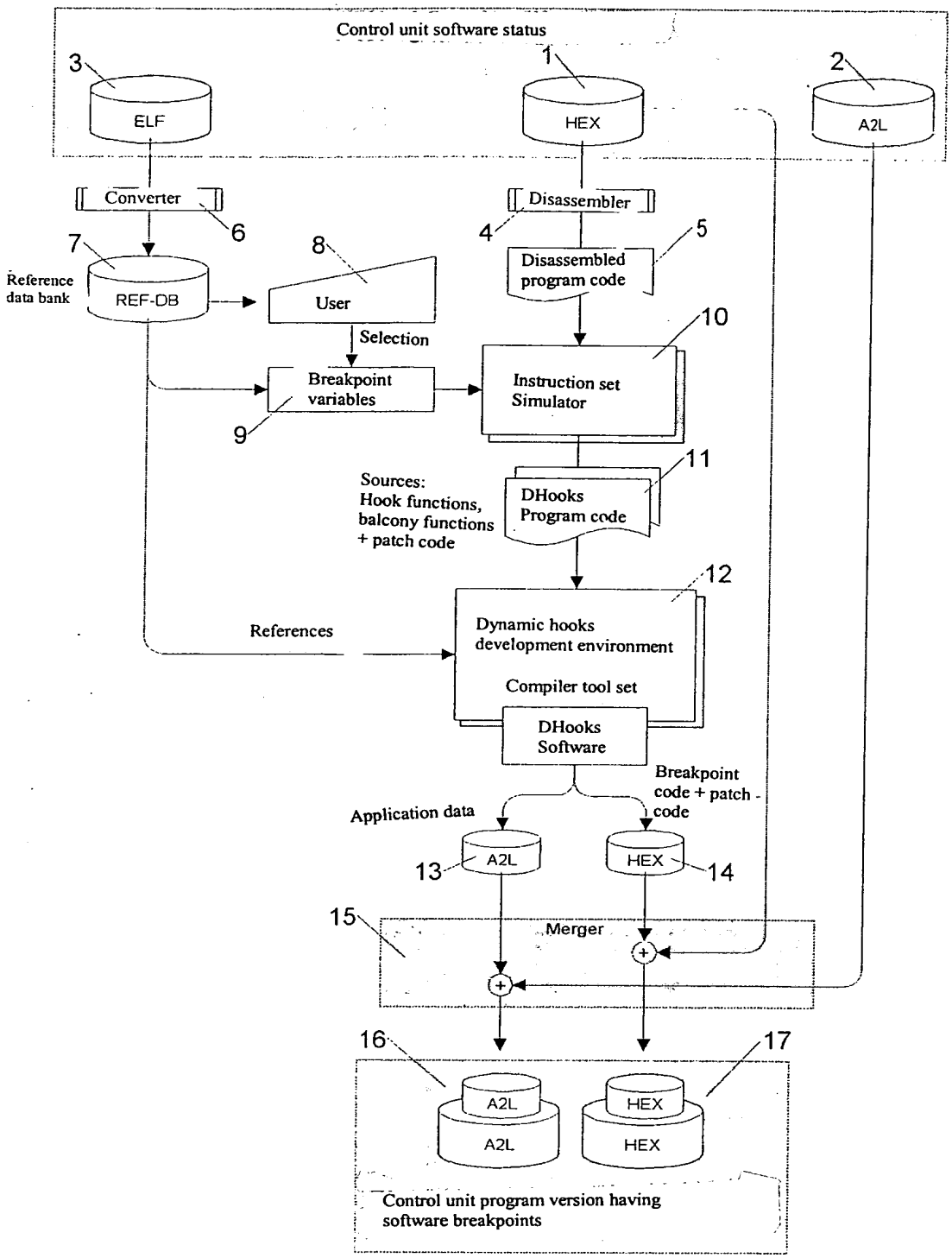


Fig. 11

BEST AVAILABLE COPY